

# Lec 6 compiler

→ left most derivation

$$E \Rightarrow \underline{E} + E \Rightarrow E * \underline{E} + E \Rightarrow id * \underline{E} + \underline{E}$$

~~identity~~ identity

$$id * \underline{id} + E \xrightarrow{\text{identity}}$$

→ Right most derivation

$$E \Rightarrow E + \underline{E} \Rightarrow E + id \Rightarrow E * \underline{E} + id \Rightarrow E * id + id \Rightarrow id * id + id$$

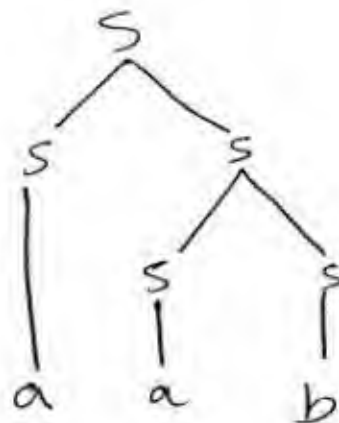
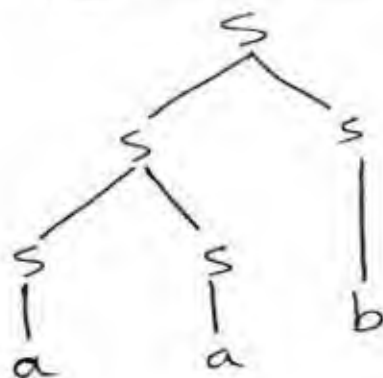
• left most • Right most • left ~ right

2-distinct

indistinct. Parsing trees

→ Ambiguity

$\Rightarrow aab$

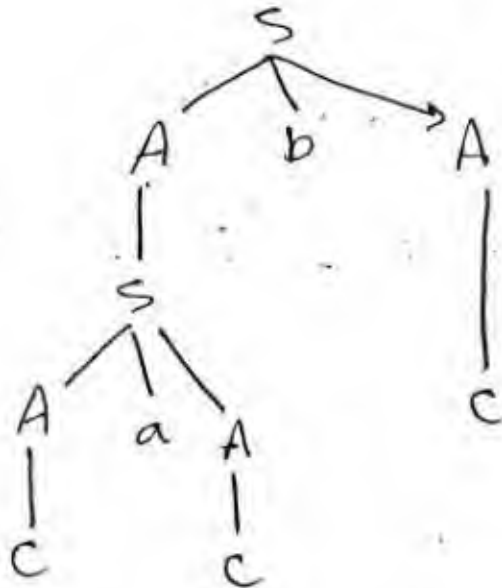
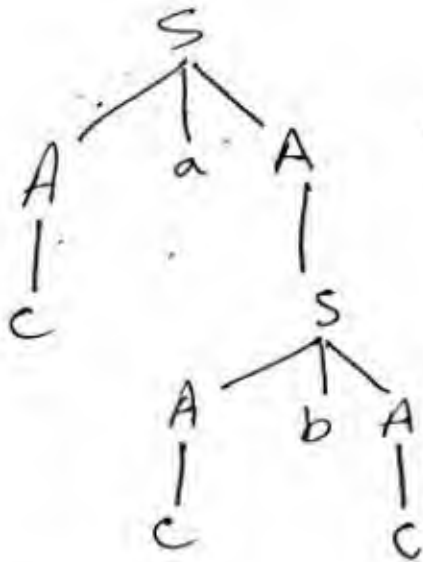


•  $E$  ليس سبب (2-distinct parsing) ←

**Ex**

a)  $S \rightarrow AaA \mid AbA$

$A \rightarrow c \mid s$



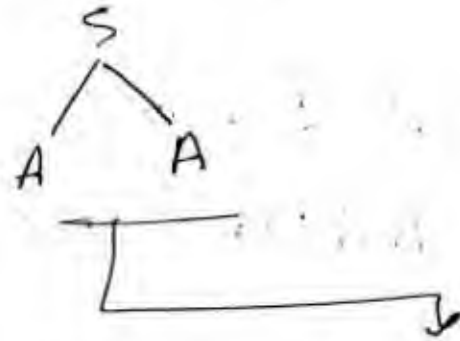
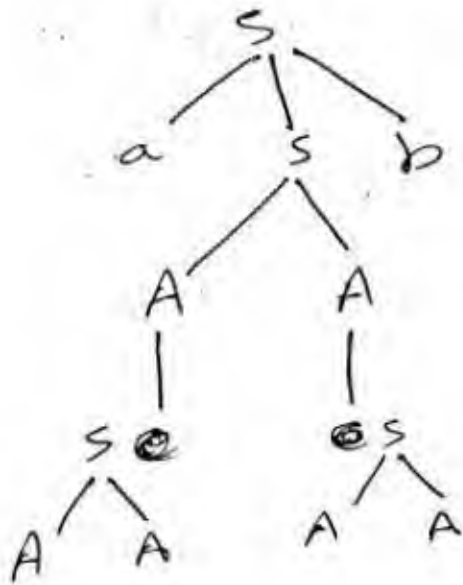
~~caccbc~~

~~caccbc~~

caccbc

Ambiguous

b)  $s \rightarrow asb$   
 $s \rightarrow AA$   
 $A \rightarrow c|s$

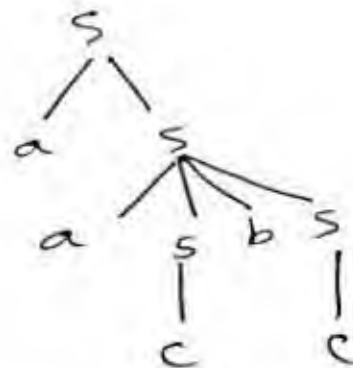
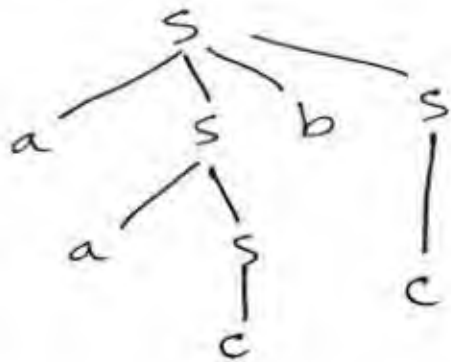


ده يا اما صيد اب ح داغ  
 از هينتي ب ح  
 وده مش (supported) في الثاني

ده صيد اب ح و صيد ب  
 وفي الوسط حاجات كثيره

UN Ambiguous

c)  $s \rightarrow asbs$   
 $s \rightarrow as$   
 $s \rightarrow c$



Ambiguous — نفس النتيجة بشكل مختلف

— حاد متعقدش ال (Grammer) و در علی اقربا  
طريقة توصلة.

— مقيس حلول ثابتة عشاه تمنع وجود ال Ambiguous

بس الحل الاقربا انك تعمل rewrite ال (grammer)  
بعثت تعمل (avoid) ال Ambiguous.

— رقم a في الأمثلة السابقة

$$S \rightarrow AaA \mid AbA$$

$$A \rightarrow c \mid s$$

— نستبدل  $B \neq AbA$  ونكون

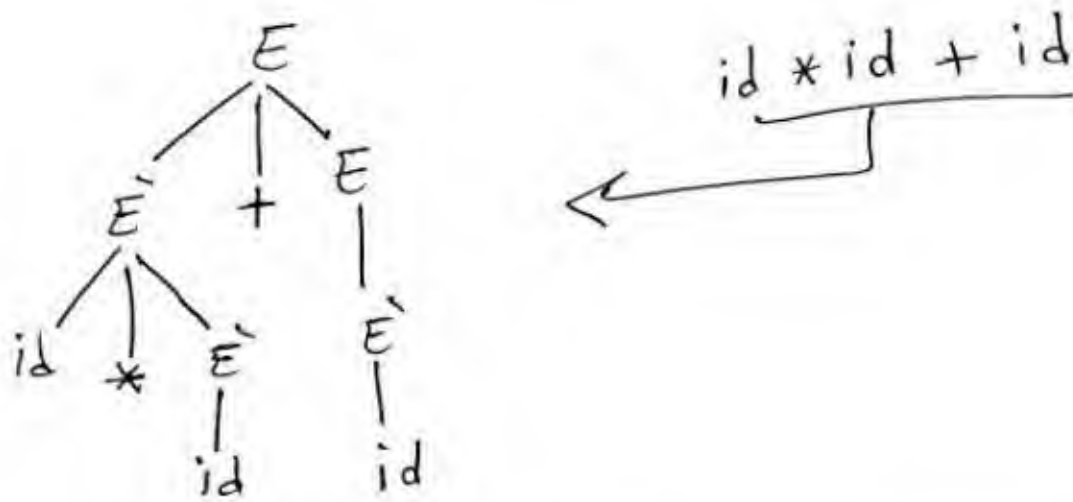
$$S \rightarrow AaA \mid B$$

$$B \rightarrow AbA$$

$$A \rightarrow c \mid s$$

$E \rightarrow E + E \rightarrow E + E + E \rightarrow E + E + E + E \rightarrow \dots$   
 $\dots \rightarrow id + id + id + \dots + id$

← لا يوجد طريقة أخرى لعملها.



← خلية واحدة ال (Non terminal) هو ال Control  
 ال (Plusing) والآخر هو ال Control ال (Timing)  
 ال الهزب.

~~if expr then MIF~~

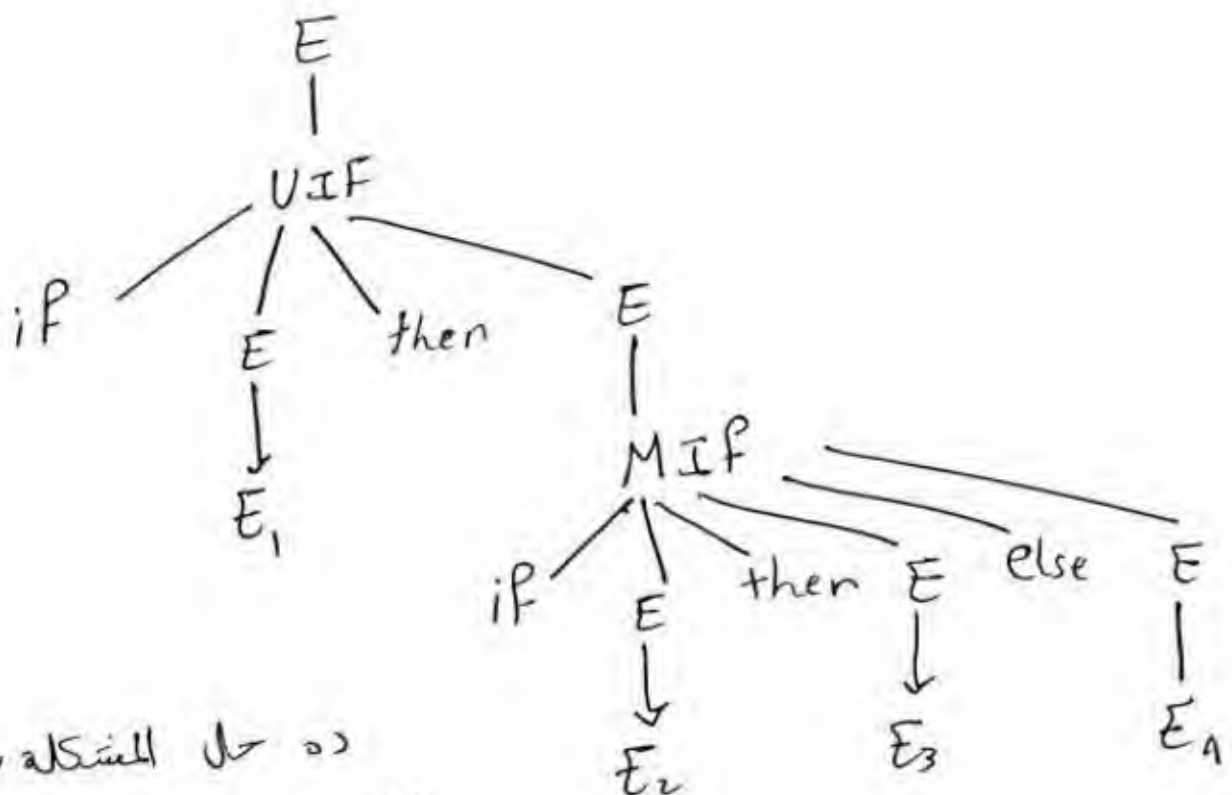
MIF  $\Rightarrow$  if E then MIF else MIF  
                     /                    /  
                 if-then else          if then else

if E then E else E

$\hookrightarrow$  if E then UIF else MIF  
 $\downarrow$   $\downarrow$   
 if then مستعار فینده  
تبعی if  
دو مستحق قبول

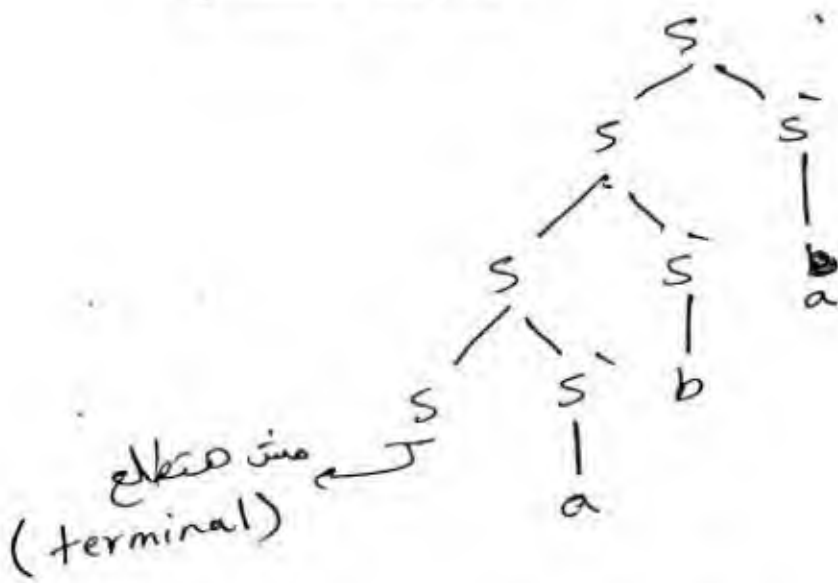
$\rightarrow$  if E then MIF else UIF  
 $\swarrow$   $\downarrow$   $\searrow$   
if then else

$\leftarrow$  دو مستحق عادی.

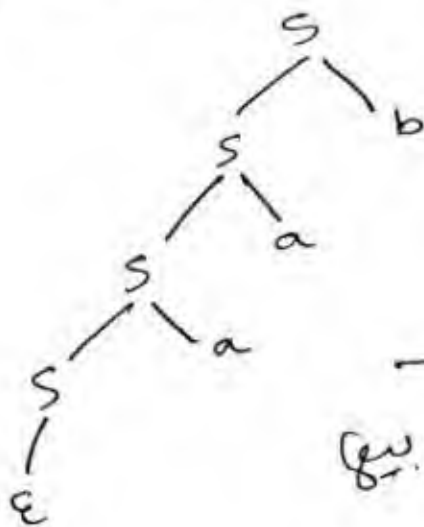


$$\begin{aligned} b) \quad S &\rightarrow S\hat{S} \\ \hat{S} &\rightarrow a|b \end{aligned}$$

→ un ambiguous



a)  $S \rightarrow Sa | Sb | \epsilon$



→ un ~~amb~~ambiguous

— لا توجد طريقة أخرى أرسل بيدي  
للحل ده

→ ده حل و ارد للسؤال.

$$\begin{aligned} S &\rightarrow s a | \bar{s} b \\ \bar{s} &\rightarrow s / \epsilon \end{aligned}$$

بالإضافة لوظائف ال (Compiler) العادية → يعمل (handling) لا (non-valid Program or. (errors))

Panic mode

→ يعمل تجاهل لا (tokens) حتى يصل إلى نقطة واضحة تجعل الجملة صحيحة.

[Ex]

$$(1 + + 2) + 3$$

→ نتمش مع المثال للبحث  
مفرد جداً عمل تجاهل لا + الثانية حتى نصل ل 2 و بكل الجملة ويعطى رسالة لا فيه (Syntax error) في المكان معين.

→ هو يعمل تجاهل حتى يصل إلى حاجة تخليه بكل جملة

$$1 + \xrightarrow{\text{حتى نصل ل}} 2$$







PLI C  
compilers

يدعم فكرة (Automatic Correction)  
in error handling  
↳ Report optional.

↳ show How to support error handling  
↳ examples for this

9